

Contents

1	Shift-Invariant Linear Systems	5
2	Display Calibration	13
3	Classification	23
4	Signal Estimation: A Geometric View	31
5	Motion Flow Field Calculation	39

List of Figures

1.1	Cosinusoidal and Sinusoidal terms in the DFS	10
2.1	Television Display	14
2.2	Frame-buffer control of Phosphor Intensity.	16
2.3	Monitor Matrix	18
3.1	Bayes Classifier for Intensity Discrimination	26
3.2	Geisler Bayesian classifier	29
4.1	Dot Product Geometry in two dimensions	32
4.2	Dot product geometry in three dimensions	33
4.3	Estimation of the Signal from the Responses	34
5.1	Perspective Calculation of Motion Flow Field	41

Chapter 1

Shift-Invariant Linear Systems

Many of the ideas in this book rely on properties of shift-invariant linear systems. In the text, I introduced these properties without any indication of how to prove that they are true. I have two goals for this section. First, I will sketch proofs of several important properties of shift-invariant linear systems. Second, I will describe *convolution* and the *Discrete Fourier Series*, two tools that help us take advantage of these properties¹.

Shift-Invariant Linear Systems: Definitions

Shift-invariance is a system property that can be verified by experimental measurement. For example, in Chapter ?? I described how to check whether the optics of the eye is shift-invariant by the following measurements. Image a line through the optics onto the retina, and measure the linespread function. Then, shift the line to a new position, forming a new retinal image. Compare the new image with the original. If the images have the same shape, differing only by a shift in position, then the optics is shift-invariant over the measured range.

We can express the empirical property of shift-invariance using the following mathematical notation. Choose a stimulus, p_i and measure the system response, r_i . Now, shift the stimulus by an amount j and measure again. If the response is shifted by j as well, then the system may be shift-invariant. Try this experiment for many values of the shift parameter, j . If the experiment succeeds for all shifts, then the system is shift-invariant.

¹The results in this appendix are expressed using real harmonic functions. This is in contrast to the common practice of using complex notation, specifically Euler's complex exponential, as a shorthand to represent sums of harmonic functions. The exposition using complex exponentials is brief and elegant, but I believe it obscures the connection with experimental measurements. For this reason, I have avoided the development based on complex notation.

If you think about this definition as an experimentalist, you can see that there are some technical problems in making the measurements needed to verify shift-invariance. Suppose that the original stimulus and response are represented at N distinct points, $i = 1, \dots, N$. If we shift the stimulus three locations so that now the fourth location contains the first entry, the fifth the second, and so forth, how do we fill in the first three locations in the new stimulus? And, what do we do with the last three values, $N - 2, N - 1, N$, which have nowhere to go?

Theorists avoid this problem by treating the real observations as if they are part of an infinite, periodic set of observations. They assume that the stimuli and data are part of an infinite periodic series with a period of N , equal to the number of original observations. If the data are infinite, the first three entries of the shifted vector are the three values at locations $-3, -2$, and -1 . If the data are periodic with period N , these three values are the same as the values at $N - 3, N - 2$, and $N - 1$.

The assumption that the measurements are part of an infinite and periodic sequence permits the theorist to avoid the experimentalist's practical problem. The assumption is also essential for obtaining several of the simple closed-form results concerning the properties of shift-invariant systems. The assumption is not consistent with real measurements since real measurements cannot be made using infinite stimuli: there is always a beginning and an end to any real experiment. As an experimentalist you must always be aware that many theoretical calculations using shift-invariant methods are not valid near the boundaries of data sets, such as near the edge of an image.

Suppose we refer to the finite input as, \mathbf{p} , and the measured output, \mathbf{l} , is finite. In the theoretical analysis we extend both of these functions to be infinite and periodic. We will use a hat symbol to denote the extended functions,

$$\hat{l}_{i+N} = l_i \quad \text{and} \quad \hat{p}_{i+N} = p_i. \quad (1.1)$$

The extended functions $\hat{\mathbf{l}}$ and $\hat{\mathbf{p}}$ agree with our measurements over the measurement range from $1, \dots, N$. By the periodicity assumption, the values outside of the measurement range are filled in by looking at the values within the measurement range. For example,

$$(\dots, \hat{l}_{-1} = \hat{l}_{N-1}, \hat{l}_0 = \hat{l}_N, \hat{l}_1, \dots, \hat{l}_N, \hat{l}_{N+1} = \hat{l}_1, \dots).$$

Convolution

Next, we derive some of the properties of linear shift-invariant systems. We begin by describing these properties in terms of the *system matrix* (see Chapter ??). Then, we will show how the simple structure of the shift-invariant system matrix permits us to relate the input and output by a summation formula called *cyclic convolution*. The

convolution formula is so important that shift-invariant systems are sometimes called *convolution systems*².

In Chapter ?? I reviewed how to measure the system matrix of an optical system for one-dimensional input stimuli. We measure the image resulting from a single line at a series of uniformly spaced input locations. If the system is shift-invariant, then the columns of the system matrix are shifted copies of one another (except for edge artifacts). To create the system matrix, we extend the inputs and outputs to be periodic functions (Equation 1.1). Then, we select a central block of size $N \times N$ to be the system matrix and we use the corresponding entries of the extended stimulus. For example, if the input stimulus consists of six values, $\mathbf{p} = (0, 0, 0, 1, 0, 0)$, and the response to this stimulus is the vector, $\mathbf{l} = (0.0, 0.3, 0.6, 0.2, 0.1, 0.0)$, then the 6×6 system matrix is

$$\hat{\mathbf{C}} = \begin{pmatrix} 0.2 & 0.6 & 0.3 & 0.0 & 0.0 & 0.1 \\ 0.1 & 0.2 & 0.6 & 0.3 & 0.0 & 0.0 \\ 0.0 & 0.1 & 0.2 & 0.6 & 0.3 & 0.0 \\ 0.0 & 0.0 & 0.1 & 0.2 & 0.6 & 0.3 \\ 0.3 & 0.0 & 0.0 & 0.1 & 0.2 & 0.6 \\ 0.6 & 0.3 & 0.0 & 0.0 & 0.1 & 0.2 \end{pmatrix}. \quad (1.2)$$

For a general linear system, we calculate the output using the summation formula for matrix multiplication in Equation ??,

$$\hat{r}_i = \sum_{j=1}^N \hat{C}_{ij} \hat{p}_j. \quad (1.3)$$

When the linear system is shift-invariant system, this summation formula simplifies for two reasons. First, because of the assumed periodicity, the summation is precisely the same when we sum over any N consecutive integers. It is useful to incorporate this generalization into the summation formula as

$$\hat{r}_i = \sum_{j=\langle N \rangle} \hat{C}_{ij} \hat{p}_j, \quad (1.4)$$

where the notation $j = \langle N \rangle$ means that summation can take place over any N consecutive integers. Second, notice that for whichever $N \times N$ block of values we choose, the typical entry of the system matrix will be

$$\hat{C}_{ij} = \hat{l}_{i-j}. \quad (1.5)$$

We can use this relationship to simplify the summation further,

$$\hat{r}_i = \sum_{j=\langle N \rangle} \hat{l}_{i-j} \hat{p}_j. \quad (1.6)$$

²Since we use only cyclic convolution here, I will drop the word “cyclic” and refer to the formula simply as convolution. This is slightly abusive, but conforms to common practice in many fields.

In this form, we see that the response depends only on the input and the linespread. The summation formula in Equation 1.6 is called *cyclic convolution*. Hence, we have shown that to compute the response of a shift-invariant linear system to any stimulus, we need measure only the linespread function.

Convolution and Harmonic Functions

Next, we study some of the properties of the convolution formula. Most important, we will see why harmonic functions have a special role in the analysis of convolution systems.

Beginning with our analysis of optics in Chapter ??, we have relied on the fact that the response of a shift-invariant system to a harmonic function at frequency f is also a harmonic function at f . In that chapter, the result was stated in two equivalent ways.

1. If the input is a harmonic at frequency f , the output is a shifted and scaled copy of the harmonic.
2. The response to a harmonic at frequency f will be the weighted sum of a sinusoid and cosinusoid at the same frequency (Equation ??).

We can derive this result from the convolution formula. Define a new variable, $k = i - j$, and substitute k into Equation 1.6. Remember that the summation can take place over any adjacent N values. Hence, the substitution yields a modified convolution formula,

$$\hat{r}_i = \sum_{k=\langle N \rangle} \hat{l}_k \hat{p}_{i-k}. \quad (1.7)$$

Next, we use the convolution formula in Equation 1.7 to compute the response to a sinusoidal input $\sin(\frac{2\pi f j}{N})$. From trigonometry we have that

$$\sin(\frac{2\pi f(i+j)}{N}) = \sin(\frac{2\pi f i}{N})\cos(\frac{2\pi f j}{N}) + \sin(\frac{2\pi f j}{N})\cos(\frac{2\pi f i}{N}). \quad (1.8)$$

Substitute Equation 1.8 into Equation 1.7, remembering that $\sin(-k) = -\sin(k)$ and $\cos(-k) = \cos(k)$.

$$\begin{aligned} \hat{r}_i &= \sum_{k=\langle N \rangle} \hat{l}_k \sin(\frac{2\pi f i}{N})\cos(\frac{2\pi f k}{N}) + \sum_{k=\langle N \rangle} \hat{l}_k \sin(-\frac{2\pi f k}{N})\cos(\frac{2\pi f i}{N}) \\ &= \sin(\frac{2\pi f i}{N}) \sum_{k=\langle N \rangle} \hat{l}_k \cos(\frac{2\pi f k}{N}) - \cos(\frac{2\pi f i}{N}) \sum_{k=\langle N \rangle} \hat{l}_k \sin(\frac{2\pi f k}{N}) \end{aligned} \quad (1.9)$$

We can simplify this expression to the form

$$\hat{r}_i = a \sin(\frac{2\pi f i}{N}) - b \cos(\frac{2\pi f i}{N}) \quad (1.10)$$

where

$$\begin{aligned} a &= \sum_{k=\langle N \rangle} \hat{l}_k \cos\left(\frac{2\pi f k}{N}\right) \\ b &= \sum_{k=\langle N \rangle} \hat{l}_k \sin\left(\frac{2\pi f k}{N}\right) \end{aligned}$$

We have shown that when the input to the system is a sinusoidal function at frequency f , the output of the system is the weighted sum of a sinusoid and a cosinusoid, both at frequency f . This is equivalent to showing that when the input is a sinusoid at frequency f , the output will be a scaled and shifted copy of the input, $s_f \sin(\frac{2\pi f i}{N} + \phi_f)$ (see Equation ??). As we shall see below, it is easy to generalize this result to all harmonic functions.

The Discrete Fourier Series: Definitions

In general, when we measure the response of a shift-invariant linear system we measure N output values. When the input is a sinusoid, or more generally a harmonic, we can specify the response using only the two numbers, a and b , in Equation 1.10. We would like to take advantage of this special property of shift-invariant systems. To do so, we need a method of representing input stimuli as the weighted sum of harmonic functions.

The method used to transform a stimulus into the weighted sum of harmonic functions is called the *Discrete Fourier Transform* (DFT). The representation of the stimulus as the weighted sum of harmonic functions is called the *Discrete Fourier Series* (DFS). We use the DFS to represent an extended stimulus, \hat{p} .

$$\hat{p}_i = \sum_{f=0}^{N-1} a_f \cos\left(\frac{2\pi f i}{N}\right) + b_f \sin\left(\frac{2\pi f i}{N}\right) . \quad (1.11)$$

We are interested in that part of the extended stimulus that coincides with our measurements. We can express the relationship between the harmonic functions and the original stimulus, \mathbf{p} , using a matrix equation

$$\mathbf{p} = \mathbf{C}\mathbf{a} + \mathbf{S}\mathbf{b} \quad (1.12)$$

which has the matrix tableau form

$$\begin{pmatrix} \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{C} \end{pmatrix} \begin{pmatrix} \mathbf{a} \end{pmatrix} + \begin{pmatrix} \mathbf{S} \end{pmatrix} \begin{pmatrix} \mathbf{b} \end{pmatrix} \quad (1.13)$$

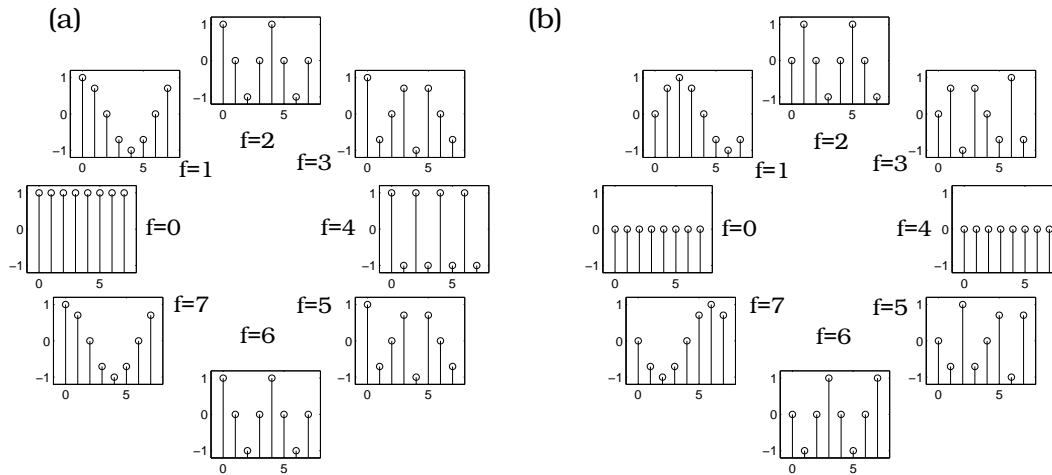


Figure 1.1: *The basis functions of the Discrete Fourier Series.* The cosinusoidal (a) and sinusoidal (b) basis functions of the discrete Fourier series representation when $N = 8$ are shown. Notice the redundancy between the functions at symmetrically placed frequencies.

The vectors \mathbf{a} and \mathbf{b} contain the coefficients a_f and b_f , respectively. The columns of the matrices \mathbf{C} and \mathbf{S} contain the relevant portions of the cosinusoidal and sinusoidal terms, $\cos(\frac{2\pi fi}{N})$ and $\sin(\frac{2\pi fi}{N})$, that are used in the DFS representation.

The DFS represents the original stimulus as the weighted sum of a set of harmonic functions (i.e., sampled sine and cosine functions). We call these sampled harmonic functions the *basis functions* of the DFS representation. The vectors \mathbf{a} and \mathbf{b} contain basis functions *weights* or *coefficients* that specify how much of each basis function must be added in to recreate the original stimulus, \mathbf{p} .

The Discrete Fourier Series: Properties

Figure 1.1 shows the sampled sine and cosine functions for a period of $N = 8$. The functions are arrayed in a circle to show how they relate to one another. There are a total of 16 basis functions. But, as you can see from Figure 1.1, they are redundant. The sampled cosinusoids in the columns of \mathbf{C} repeat themselves (in reverse order); for example, when $N = 8$ the cosinusoids for $f = 1, 2, 3$ are the same as the cosinusoids for $f = 7, 6, 5$. The sampled sinusoids in the columns of \mathbf{S} also repeat themselves except for a sign reversal (multiplication by negative one). There are only four independent sampled cosinusoids, and four independent sampled sinusoids. As a result of this redundancy, neither the \mathbf{S} nor the matrix \mathbf{C} is invertible.

Nevertheless, the properties of these harmonic basis functions make it simple to calculate the vectors containing the weights of the harmonic functions from the

original stimulus, \mathbf{p} . To compute \mathbf{a} and \mathbf{b} , we multiply the input by the basis functions, as in

$$\mathbf{a} = \frac{1}{N} \mathbf{C}^T \mathbf{p} \quad \text{and} \quad \mathbf{b} = \frac{1}{N} \mathbf{S}^T \mathbf{p}. \quad (1.14)$$

We can derive the relationship in Equation 1.14 from two observations. First, the matrix sum, $\mathbf{H} = \mathbf{S} + \mathbf{C}$ has a simple inverse. The columns of \mathbf{H} are orthogonal to one another, so that the inverse of \mathbf{H} is simply³.

$$\mathbf{H}^{-1} = \frac{1}{N} \mathbf{H}^T = \frac{1}{N} (\mathbf{S} + \mathbf{C})^T.$$

Second, the columns of the matrices \mathbf{C} and \mathbf{S} are perpendicular to one another: $\mathbf{0} = \mathbf{C} \mathbf{S}^T$. This observation should not be surprising since continuous sinusoids and cosinusoids are also orthogonal to one another.

We can use these two observations to derive Equation 1.14 as follows. Express the fact that \mathbf{H} and $\frac{1}{N} \mathbf{H}^T$ are inverses as follows:

$$\mathbf{I}_{N \times N} = \mathbf{H} \left(\frac{1}{N} \mathbf{H}^T \right) \quad (1.15)$$

$$= \frac{1}{N} (\mathbf{C} + \mathbf{S})(\mathbf{C} + \mathbf{S})^T \quad (1.16)$$

$$= \frac{1}{N} (\mathbf{C} \mathbf{C}^T + \mathbf{S} \mathbf{S}^T), \quad (1.17)$$

where $\mathbf{I}_{N \times N}$ is the identity matrix. Then, multiply both sides of Equation 1.15 by \mathbf{p} ,

$$\mathbf{p} = \frac{1}{N} (\mathbf{C} \mathbf{C}^T \mathbf{p} + \mathbf{S} \mathbf{S}^T \mathbf{p}). \quad (1.18)$$

Compare Equation 1.18 and Equation 1.12. Notice that the Equations become identical if we make the assignments in Equation 1.14. This completes the sketch of the proof.

Measuring a Convolution System using Harmonic Functions.

Finally, we show how to predict the response of a shift-invariant linear system to any stimulus using only the responses to unit amplitude cosinusoidal inputs. This result is the logical basis for describing system performance from measurements of the response to harmonic functions. When the system is linear and shift-invariant, its responses to harmonic functions is a complete description of the system; but, this is not true for arbitrary linear systems.

³This observation is the basis of another useful linear transform method called the *Hartley Transform* (Bracewell, 1986).

Because cosinusoids and sinusoids are shifted copies of one another, the response of a shift-invariant linear system to these functions is the same except for a shift. From a calculation like the one in Equation 1.8, except using a cosinusoidal input, we can calculate the following result: If the response to a cosinusoid at f is a sum of cosinusoid and sinusoid with weights, (a_f, b_f) , then, the response to a sinusoid at frequency f will have weights $(b_f, -a_f)$. Hence, if we know the response to a cosinusoid at f , we also know the response to a sinusoid at f .

Next, we can use our knowledge of the response to sinusoids and cosinusoid at f to predict the response to any harmonic function at f . Suppose that the input is a harmonic function $a\cos(\frac{2\pi fi}{N}) + b\sin(\frac{2\pi fi}{N})$, and the output is, say $a'\cos(\frac{2\pi fi}{N}) + b'\sin(\frac{2\pi fi}{N})$. If the response to a unit amplitude cosinusoid is $u_f\cos(\frac{2\pi fi}{N}) + v_f\sin(\frac{2\pi fi}{N})$, then the response to a unit amplitude sinusoid is $v_f\cos(\frac{2\pi fi}{N}) - u_f\sin(\frac{2\pi fi}{N})$. Using these two facts and linearity we calculate the coefficients of the response,

$$\begin{aligned} a' &= au_f + bv_f \\ b' &= av_f - bu_f. \end{aligned} \tag{1.19}$$

We have shown that if we measure the system response to unit amplitude cosinusoidal inputs, we can compute the system response to an arbitrary input stimulus as follows.

1. Compute the DFS coefficients of the input stimulus using Equation 1.14.
2. Calculate the DFS coefficients of the output using Equation 1.19.
3. Reconstruct the output from the coefficients using Equation 1.11.

You will find this series of calculations used implicitly at several points in the text. For example, we followed this organization when I described measurements of the optical quality of the lens (Chapter ??) and when I described measurements of behavioral sensitivity to spatiotemporal patterns (Chapter ??).

Chapter 2

Display Calibration

Visual displays based on a *cathode ray tube* (CRT) are used widely in business, education and entertainment. The CRT reproduces color images using the principles embodied in the color-matching experiment (Chapter ??).

The design of the color CRT is one of the most important applications of vision science; thus, it is worth understanding the design as an engineering achievement. Also, because the CRT is used widely in experimental vision science, understanding how to control the CRT display is an essential skill for all vision scientists. This appendix reviews several of the principles of monitor calibration.

An Overview of a CRT Display

Figure 2.1a shows the main components of a color CRT display. The display contains a *cathode*, or *electron gun*, that provides a source of electrons. The electrons are focused into a beam whose direction is deflected back and forth in a raster pattern so that it scans the faceplate of the display.

Light is emitted by a process of absorption and emission that occurs at the faceplate of the display. The faceplate consists of a phosphor painted onto a glass substrate. The phosphor absorbs electrons from the scanning beam and emits light. A signal, generally controlled from a computer, modulates the intensity of the electron beam as it scans across the faceplate. The intensity of the light emitted by the phosphor at each point on the faceplate depends on the intensity of the electrons beam as it scans past that point.

Monochrome CRTs have a single electron beam and a single type of phosphor. In monochrome systems the control signal only influences the intensity of the phosphor emissions. Color CRTs use three electron beams; each stimulates one of three phosphors. Each of the phosphors emits light of a different spectral power

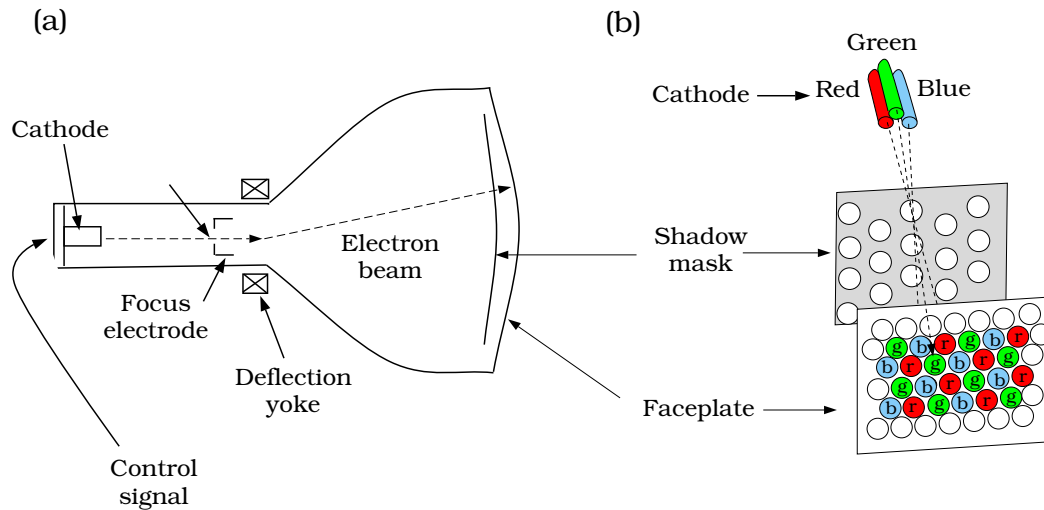


Figure 2.1: *Overview of a cathode ray tube display.* (a) A side view of the display showing the cathode, which is the source of electrons, and a method of focusing the electrons into a beam that is deflected in a raster pattern across the faceplate of the display. (b) The geometrical arrangement of the electron beams, shadow-mask, and phosphor allows each electron beam to stimulate only one of the three phosphors.

distribution. By separately controlling the emissions of the three types of phosphors, the user can vary the spectral composition of the emitted light. The light emitted from the CRT is always the mixture of three primary lights from the three phosphors, usually called the red, green and blue phosphors.

In order that each electron beam stimulate emissions from only one of the three types of phosphors, a metal plate, called a *shadow-mask*, is interposed between the three electron guns and the faceplate. A conventional shadow-mask is a metal plate with a series of finely spaced holes. The relative positions of the holes and the electron guns are arranged, as shown in Figure 2.1b, so that as the beam sweeps across the faceplate the electrons from a single gun that pass through a hole are absorbed by only one of the three types of phosphors; electrons from that gun that would have stimulated the other phosphors are absorbed or scattered by the shadow-mask.

The frame-buffer

In experiments, the control signal sent to the CRT display is usually created using computer software. There are two principal methods for creating these control signals. In one method, the user controls the three electron beam intensities by writing out the values of three matrices into three separate *video frame-buffers*. Each matrix specifies the control signal for one of the electron guns. Each matrix entry specifies the desired voltage level of the control signal at a single point on the

faceplate. Usually, the intensity levels within each matrix are quantized to 8 bits, so this computer display architecture is called *24 bit color*, or *RGB color*.

In a second method, the user writes a single matrix into a single frame-buffer. The value at each location in this matrix is converted into three control signals sent to the display according to a code contained in a *color look-up table*. This architecture is called *indexed color*. This method is cost-effective because it does away with two of the three frame-buffers. When using this method, the user can only select among 256 (8 bits) colors when displaying a single image.

The Display Intensity

Calibrating a visual display means measuring the relationship between the frame-buffer values and the light emitted by the display. In this section we will discuss the relationship between the frame-buffer values and the intensity of the emitted light. In the next section we will discuss the relationship between the frame-buffer values and the spectral composition of the emitted light.

We can measure the relationship between the value of a frame-buffer entry and the intensity of the light emitted from the display as follows: Set the frame buffer entries controlling one of the three phosphor display intensities, say within a rectangular region, to a single value. Measure the intensity of the light emitted from the displayed rectangle. Repeat this measurement at many different frame-buffer values for this phosphor, and then for the other two phosphors.

The dashed curve in Figure 2.2 measures the ratio of the intensity at the highest frame-buffer level to the intensity at each of the other frame-buffer levels for the green phosphor. We can summarize the difference using this single ratio, the *relative intensity* because for over most of the range the spectral power distribution of the light emitted at one frame-buffer level is the same as the spectral power distribution of the light emitted from the monitor at maximum except for a scale factor. The insets in the graph show two examples of the spectral power distribution, measured when the frame-buffer was set to 255 and 190. These two curves have the same overall shape; they differ by a scale factor of one half.

We can approximate the curve relating the relative intensity of the light emitted from this CRT display, I , and the frame-buffer value, v , by a function of the form

$$I = \alpha v^\gamma + \beta,$$

where α and β are two fitting parameters. For most CRTs, the exponent of the power function, γ , has a value near 2.2 (see Brainard, 1989; Berns et al., 1993ab).

The nonlinear function relating the frame-buffer values and the relative intensity is due to the physics of the CRT display. While nonlinear functions are usually viewed

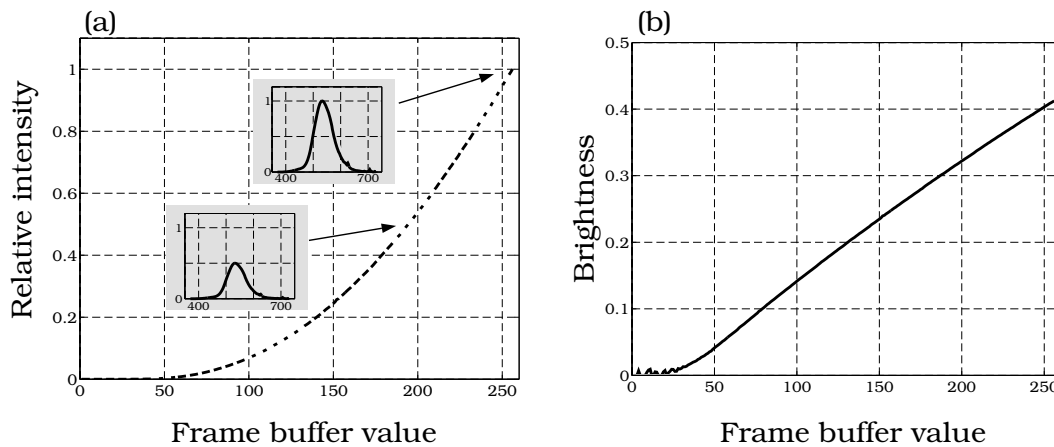


Figure 2.2: *Frame-buffer value and display intensity*. (a) The dashed curve measures the intensity of the emitted light relative to the maximum intensity. The data shown are for the green phosphor. The insets in the graph show the complete spectral power distribution of the light at two different frame-buffer levels. (b) The dashed curve describing the relative intensities is replotted, using Stevens' Power Law, to show the linear relationship between the frame-buffer value and perceived brightness.

with some dread, this particular nonlinear relationship is desirable because most users want the frame-buffer values to be linear with the brightness; they don't care how the frame-buffer values relate to intensity. As it turns out, perceived brightness, B , is related to intensity through a power law relationship called *Stevens' Power Law* (Stevens, 1962; Goldstein, 1989; Sekuler and Blake, 1985), namely,

$$B = aI^{0.4}$$

where a is a fitting parameter. Somewhat fortuitously, the nonlinear relationship between frame-buffer values and intensity compensates for the nonlinear relationship between intensity and brightness. To show this, I have replotted Figure 2.2a on a graph whose vertical scale is brightness, that is relative intensity raised to the 0.4 power. This graph shows that the relationship between the frame-buffer value and brightness is nearly linear. This has the effect of equalizing the perceptual steps between different levels of the frame-buffer and simplifying certain aspects of controlling the appearance of the display.

The Display Spectral Power Distribution

The spectral power distribution of the light emitted in each small region of a CRT display is the mixture of the light emitted by the three phosphors. Since the mixture of lights obeys superposition, we can characterize the spectral power distributions emitted by a CRT using simple linear methods.

Suppose we measure the spectral power distribution of the red, green and blue phosphors at their maximum intensity levels. We record these measurements in three column vectors, \mathbf{m}_i , $i = 1, 2, 3$.

By superposition, we know that light from the monitor screen is always the weighted sum of light from the three phosphors. For example, if all three of the phosphors are set to their maximum levels, the light emitted from the screen, call it \mathbf{t} , will have a spectral power distribution of

$$\mathbf{m}_1 + \mathbf{m}_2 + \mathbf{m}_3.$$

More generally, if we set the phosphors to the three relative intensities $\mathbf{e} = (e_r, e_g, e_b)$, the light emitted by the CRT will be the weighted sum

$$\mathbf{t} = e_r \mathbf{m}_1 + e_g \mathbf{m}_2 + e_b \mathbf{m}_3$$

Figure 2.3 is a matrix tableau that illustrates how to compute the spectral power distribution of light emitted from a CRT display. The vector \mathbf{e} contains the relative intensity of each of the three phosphors. The three columns of a matrix, call it \mathbf{M} , contain the spectral power distributions of the light emitted by the red, green and blue phosphors at maximum intensity. The spectral power distribution of light emitted from the monitor is the product $\mathbf{M}\mathbf{e}$.

The light emitted from a CRT is different from lights we encounter in natural scenes. For example, the spectral power distribution of the red phosphor, with its sharp and narrow peaks, is unlike the light we see in natural images. Nonetheless, we can adjust the three intensities of the three phosphors on a color CRT to match the appearance of most spectral power distributions, just as we can match appearance in the color-matching experiment by adjusting the intensity of three primary lights (see Chapter ??).

The Calibration Matrix

In many types of psychophysical experiments, we must be able to specify and control the relative absorption rates in the three types of cones. In this section, I show how to measure and control the relative cone absorptions when the phosphor spectral power distributions and the relative cone photopigment spectral sensitivities are known.

We use two basic matrices in these calculations. We have already created the matrix \mathbf{M} whose three columns contain the spectral power distributions of the phosphors at maximum intensity. We also create a matrix, \mathbf{B} , whose three columns contain the cone absorption sensitivities (L, M, and S), measured through the cornea and lens

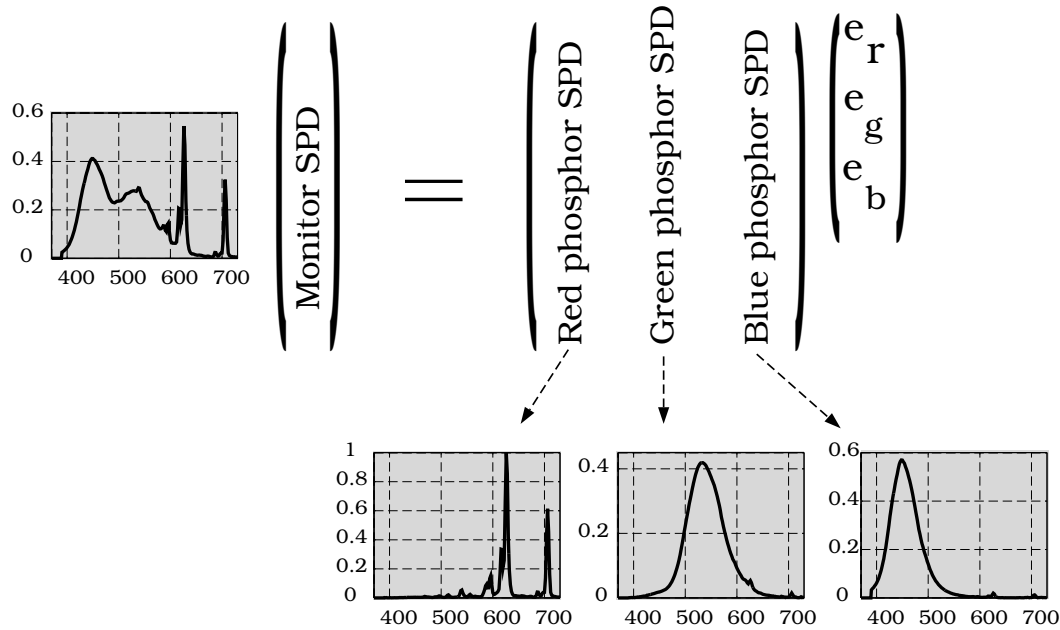


Figure 2.3: The spectral power distribution of light emitted from a CRT. The entries $\mathbf{e} = (e_r, e_g, e_b)$ are the relative intensity of the three phosphors. The columns of \mathbf{M} contain the spectral power distributions at maximum intensity. The vector calculated by $\mathbf{M}\mathbf{e}$ is the spectral power distribution of the light emitted by the CRT. The output shown in the figure was calculated for $\mathbf{e} = (0.5, 0.6, 0.7)^T$.

(see Table ??.) Given a set of relative intensities, e , we calculate the relative cone photopigment absorption rates, r by the matrix product¹

$$r = B^T M e = H e. \quad (2.1)$$

We call the matrix $H = B^T M$, the monitor's *calibration matrix*. This matrix relates the linear phosphor intensities to the relative cone absorption rates.

As an example, I have calculated the calibration matrix for the monitor whose phosphor spectral power distributions are shown in Figure 2.3, and for the cone absorption spectra listed in Table ?. The resulting calibration matrix is

$$\begin{pmatrix} 0.2732 & 0.9922 & 0.1466 \\ 0.1034 & 0.9971 & 0.2123 \\ 0.0117 & 0.1047 & 1.0000 \end{pmatrix}$$

Each column of the calibration matrix describes the relative cone absorptions caused by emissions from one of the phosphors: The absorptions from the red phosphor are in the first column, and absorptions due to the green and blue phosphors are in the second and third columns, respectively.

Suppose the red phosphor stimulated only the L cones, the green the M cones, and the blue the S cones. In that case, the calibration matrix would be diagonal, and we could control the absorptions in a single cone class by adjusting only one of the phosphor emissions. In practice, the light from each of the CRT phosphors is absorbed in all three cone types and the calibration matrix is never diagonal. As a result, to control the cone absorptions we must take into account the effect each phosphor has on each of the three cone types. This complexity is unavoidable because of the overlap of the cone absorption curves and the need to use phosphors with broadband emissions to create a bright display. Consequently, the calibration matrix is never diagonal.

Example Calculations

Now, we consider two ways we might use the calibration matrix. First, we might wish to calculate the receptor absorptions to a particular light from the CRT. Second, we might wish to know how to adjust the relative intensities of the CRT in order to achieve a particular pattern of cone absorptions.

Using the methods described so far, you can calculate the relative cone absorption rates for any triplet of frame-buffer entries. Suppose the frame-buffer values are set

¹This calculation applies to spatially uniform regions of a display, in which we can ignore the complications of chromatic aberration. Marimont and Wandell (1994) describe how to include the effects of chromatic aberration.

to (128, 128, 0). The pattern will look yellow since only the red and green phosphors are excited. Assuming that the curves relating relative intensity to frame-buffer level are the same for all three phosphors (Figure 2.2), we find that the relative intensities will be $\mathbf{e} = (0.1524, 0.1524, 0.0)^T$. The product $\mathbf{H}\mathbf{e}$ yields the relative cone absorption rates $\mathbf{r} = (0.1929, 0.1677, 0.0177)^T$. If we set the frame-buffer to (128, 128, 128), which appears gray, the relative intensities are (0.1524, 0.1524, 0.1524) and the relative cone absorption rates are $\mathbf{r} = (0.2152, 0.2001, 0.1702)^T$.

A second common type of calculation is used in behavioral experiments in which the experimenter wants to create a particular pattern of cone absorptions. To infer the relative display intensities necessary to achieve a desired effect on the cone absorptions, we must use the inverse of the calibration matrix, since

$$\mathbf{e} = \mathbf{H}^{-1}\mathbf{r}.$$

To continue our example, the inverse of the calibration matrix is

$$\mathbf{H}^{-1} = \begin{pmatrix} 5.8677 & -5.8795 & 0.3876 \\ -0.6074 & 1.6344 & -0.2579 \\ -0.0049 & -0.1025 & 1.0225 \end{pmatrix}$$

Suppose we wish to display a pair of stimuli that differ only in their effects on the S cones. Let's begin with a stimulus whose relative intensities are $\mathbf{e} = (0.5, 0.5, 0.5)^T$. Using the calibration matrix, we calculate that the relative cone absorption rates from this stimulus: they are (.706, .656, .558). Now, let's create a second stimulus with a slightly higher rate of S cone absorptions: say, $\mathbf{r} = (.706, .656, .700)$. Using the inverse calibration matrix, we can calculate the relative display intensities needed to produce the second stimulus: they are (0.555, 0.4634, 0.645).

Notice that to create a stimulus difference seen only by the S cones, we needed to adjust the intensities of all three phosphor emissions. For example, to increase the S cone absorptions we need to increase the intensity of the blue phosphor emissions. This will also cause some increases in the L and M cone absorptions, and we must compensate for this by decreasing the emissions from the red and green phosphors.

Color Calibration Tips

For most of us calibration is not, in itself, the point. Rather, we calibrate displays to describe and control experimental stimuli. If your experiments only involve a small number of stimuli, then it is best to calibrate those stimuli exhaustively. This strategy avoids making a lot of unnecessary assumptions, and your accuracy will be limited only by the quality of your calibration instrument.

In some experiments, and in most commercial applications, the number of stimuli is too large for exhaustive calibration. Brainard (1989) calculates that for the most

extreme case, in which one has a 512×512 spatial array with RGB buffers at 8 bits of resolution there are $10^{1,893,917}$ different stimulus patterns. So many patterns, so little time.

Because of the large number of potential stimuli, we need to build a model of the relationship between the frame-buffer entries and the monitor output. The discussion of calibration in this appendix is based on an implicit model of the display. To perform a high quality calibration, you should check some of these assumptions. What are these implicit assumptions, and how close will they be to real performance?

Spatial Independence. First, we have assumed that the transfer function from the frame buffer values to the monitor output is independent of the spatial pattern in the frame buffer. Specifically, we have assumed that the light measurements we obtain from a region are the same if the surrounding area were set to zero or set to any other intensity level.

In fact, the intensity in a region is not always independent of the spatial pattern in nearby regions (see e.g. Lyons and Farrell, 1989, Naiman and Makous, 1992). It can be very difficult and impractical to calibrate this aspect of the display. As an experimenter, you should choose your calibration conditions to match the conditions of your experiment as closely as possible so you don't need to model the variations with spatial pattern. For example, if your experiments will use rectangular patches on a gray background, then calibrate the display properties for rectangular patches on a gray background, not on a black or white background.

If you are interested in a harsh test to evaluate spatial independence of a display, try displaying a square pattern consisting of alternating pixels with 0 and 255. When you step away from the monitor, the pattern will blur; in principle, the brightness of the this pattern should match the brightness of a uniform square of the same size all of whose values are set to the frame-buffer value whose relative intensity is 0.5. You can try the test using alternating horizontal lines, or alternating vertical lines, or random dot arrays.

Phosphor Independence. Brainard (1989) points out that the spatial independence assumption reduces the number of measurements to approximately 1.6×10^7 . Still too many measurements to make; but, at least we have reduced the problem so that there are fewer measurements than the number of atoms in the universe.

It is our second assumption, *phosphor independence*, that makes calibration practical. We have assumed that the signals emitted from three phosphors can be measured independently of one another. For example, we measured the relative intensities for the red phosphor and assumed that this curve is the same no matter what the state

of the green and blue phosphor emissions. The phosphor independence assumption implies that we need to make only 3×256 measurements, the relative intensities of each of the three phosphors, to calibrate the display. (Once the spectral power distributions of the phosphors are known; or equivalently, the entries of the calibration matrix).

Before performing your experiments, it is important to verify phosphor independence. Measure the intensity of the monitor output to a range of, say, red phosphor values when green frame-buffer is set to zero. Then measure again when the green frame-buffer is set to its maximum value. The relative intensities of the red phosphor you measure should be the same after you correct for the additive constant from the green phosphor. In my experience, this property does fail on some CRT displays; when it fails your calibration measurements are in real trouble. I suspect that phosphor independence fails when the power supply of the monitor is inadequate to supply the needs of the three electron guns. Thus, when all three electron guns are being driven at high levels, the load on the power supply exceeds the compliance range and produces a dependence between the output levels of the different phosphors. This dependence violates the assumptions of our calibration procedure and makes calibration of such a monitor very unpleasant. Get a new monitor.

For further discussion of calibration, consult some papers in which authors have described their experience with specific display calibration projects (e.g. Brainard, 1989; Cowan and Rowell, 1986; Post, 1989; Berns et al., 1993ab).

Chapter 3

Classification

Many visual judgments are classifications: an edge is present, or not; a part is defective, or not; a tumor is present, or not. The threshold and discrimination performances reviewed in Chapter ?? are classifications as well: I saw it, or not; the stimulus was this one, or that. This appendix explains some of the basic concepts in classification theory and their application to understanding vision.

We will explore the issues in classification by analyzing some simple behavioral decisions, the kind that take place in many vision experiments. Suppose that during an experimental trial, the observer must decide which of two visual stimuli, A or B , is present. Part of the observer's decision will be based on the pattern of cone absorptions during the experimental trial. We can list the cone absorptions in a vector, \mathbf{d} , whose values represent the number of absorptions in each of the cones. Because there are statistical fluctuations in the light source, and variability in the image formation process, the pattern of cone absorptions created by the same stimulus varies from trial to trial. As a result, the pattern of cone absorptions from the two different stimuli may sometimes be the same, and perfect classification may be impossible.

What response strategy should the subject use to classify the stimuli correctly as often as possible? A good way to lead your life is this: *When you are uncertain and must decide, choose the more likely alternative.*

We can translate this simple principle into a statistical procedure by the following set of calculations. Suppose that we know the probability of the signal being A when we observe \mathbf{d} . This is called the *conditional probability*, $P(A | \mathbf{d})$, which is read as "the probability of A given \mathbf{d} ." Suppose we also know the conditional probability that the signal is B is $P(B | \mathbf{d})$. The observer should decide that the signal is the one that is more likely given the data, namely

$$\text{If } P(A | \mathbf{d}) > P(B | \mathbf{d}) \text{ choose } A, \text{ else } B. \quad (3.1)$$

We call the expression in Equation 3.1 a *decision rule*. The decision rule in Equation 3.1 is framed in terms of the likelihood of the stimuli (A or B) given the data (d). This formulation of the probabilities runs counter to our normal learning experience. During training, we know that stimulus A has been presented, and we experience a collection of cone absorptions. Experience informs us about probability of the data given the stimulus, $P(d | A)$, not the probability of the stimulus given the data, $P(A | d)$. Hence, we would like to reformulate Equation 3.1 in terms of the way we acquire our experience.

Bayes Rule is a formula that converts probabilities derived from experience, $P(d | A)$, into the form we need for the decision-rule, $P(A | d)$. The expression for Bayes Rule is

$$P(A | d) = P(d | A) \frac{P(A)}{P(d)} \quad (3.2)$$

where $P(A)$ is the probability the experimenter presents signal A (in this case one-half) and $P(d)$ is the probability of observing d quanta. As we shall see, this second probability turns out to be irrelevant to our decision.

The probabilities on the right hand side of Bayes Rule are either estimated from our experience, $P(d | A)$, or they are a structural part of the experimental design $P(A)$. The probability that stimulus A or B is presented is called the *a priori* probability, or *base rate*. The probability $P(d | A)$ is called the *likelihood* of the observation given the hypothesis. The probability, $P(A | d)$ is called the *a posteriori* probability. Bayes Rule combines the base rate and the likelihood of the observation to form the *a posteriori* probability of the hypothesis.

We can use Bayes Rule to express the decision criterion in Equation 3.1 in this more convenient form.

$$\text{If } P(d | A) \frac{P(A)}{P(d)} > P(d | B) \frac{P(B)}{P(d)} \text{ choose } A, \text{ else } B. \quad (3.3)$$

Since the probability of observing the data, $P(d)$, divides both of sides of this inequality, this probability is irrelevant to the decision. Therefore, we can re-write Equation 3.3 as

$$\frac{P(d | A)}{P(d | B)} > \frac{P(B)}{P(A)}. \quad (3.4)$$

The term on the left hand side of the Equation 3.4 is called the *likelihood ratio*. The quantity on the right is called the *odds ratio*. The formula tells us that we should select A when the likelihood ratio, which depends on the data, exceeds the odds ratio, which depends on the *a priori* knowledge. A system that uses this formula to classify the stimuli is called a *Bayes classifier*.

A Bayes classifier for an Intensity Discrimination Task

In this section we will devise a Bayes classifier for a simple experimental decision. Suppose that we ask an observer to decide whether we have presented one of two brief visual stimuli; the two stimuli are identical in all ways but their intensity. We suppose that the observer decides which stimulus was presented based on the total number of cone absorptions during the trial, $\sum_{i=1}^N d_i = d$, and without paying attention to the spatial distribution of the cone absorptions.

Across experimental trials, the number of absorptions from A will vary. There are two sources of this variability. One source of variability is in the stimulus itself. Photons emitted by a light source are the result of a change in the energy level of an electron within the source material. Each electron within the material has some probability of changing states and yielding some energy in the form of a photon. This change in excitation level is a purely statistical event and cannot be precisely controlled. The variability in the number of emitted photons, then, is inherent in the physics of light emission and cannot be eliminated.

A second source of variability is in the observer. On different experimental trials, the position of the eye, accommodative state of the lens, and other optical factors will vary. As these image formation parameters vary, the chance that photons are absorbed within the photopigment will vary. These statistic fluctuations are unavoidable as well.

The physical variability is easy to model, while the biological variability is quite subtle. For this illustrative calculation, we ignore the effects of biological variability and consider only the stimulus variability. In this case, the number of absorptions will follow the *Poisson distribution*. The formula for the Poisson probability distribution describes the probability of emitting d quanta given a mean level of μ ,

$$P(d \mid \mu) = (\mu^d / d!) e^{-\mu}. \quad (3.5)$$

The value μ is called the *rate-parameter* of the Poisson distribution. The variance of the Poisson distribution is also μ .

Figure 3.1 shows the probability distributions of the number of cone absorptions from the two stimuli, A and B . For this illustration, I have assumed that the mean absorptions from the two stimuli are $\mu_A = 8$ and $\mu_B = 12$, respectively. Since the a priori stimulus probabilities are equal, the observer will select stimulus A whenever $P(d \mid A)$ exceeds $P(d \mid B)$. For this pair of distributions, this occurs whenever the observation is less than 9.

Now, suppose the experimenter presents A with probability 0.75 and B with probability 0.25. In this case, the subject can be right three quarters of the time simply by always guessing A . Given the strong a priori odds for A , the Bayes classifier will only choose B if the likelihood exceeds three to one. From the

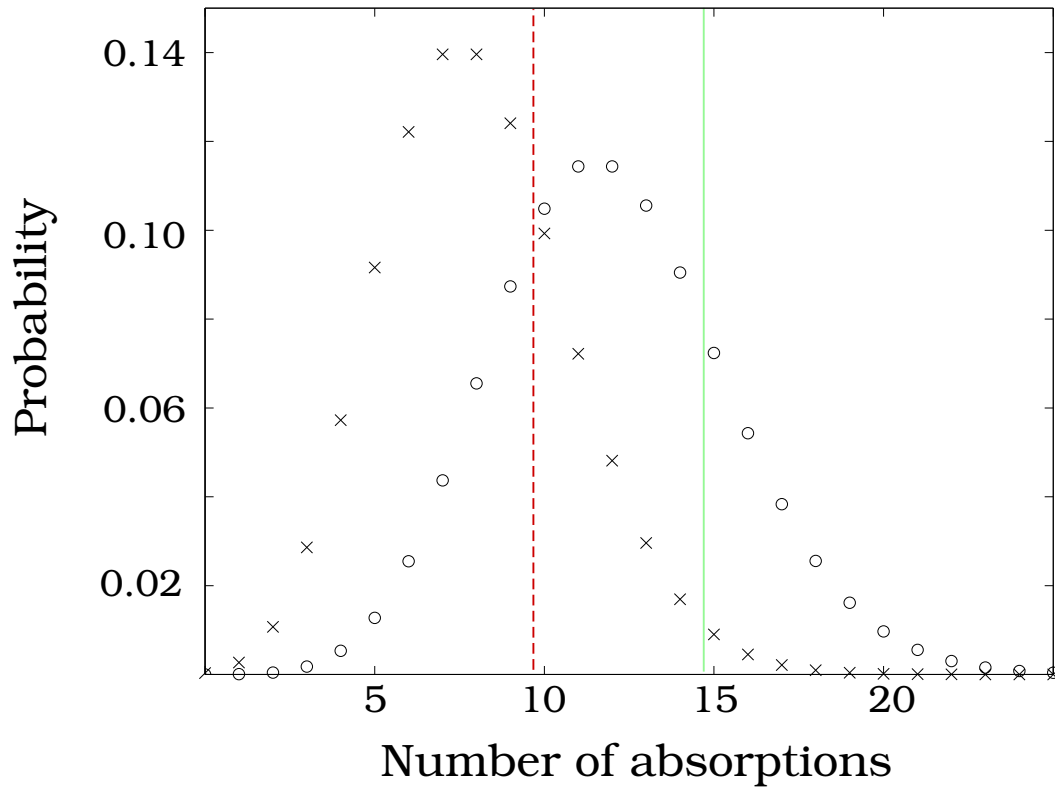


Figure 3.1: *Bayes classifier for an intensity discrimination.* The Poisson distributions of stimulus A with rate parameter $\mu_A = 8$ (x's) and and B with rate parameter $\mu_B = 12$ (o's) are shown. When the a priori probabilities of seeing the stimuli are equal, the Bayes classifier selects B when the absorptions exceed the dashed line drawn through the graph. When the a priori probabilities are $P(A) = 0.75$ and $P(B) = 0.25$, the Bayes classifier selects B when the absorptions exceed the solid line drawn through the graph.

distributions in Figure 3.1 we find this occurs when there are at least 13 absorptions. The Bayes classifier uses the data and the a priori probabilities to make a decision ¹.

A Bayes classifier for a Pattern Discrimination Task

To discriminate between different spatial patterns, or stimuli located at different spatial positions, the observer must use the spatial pattern of cone absorptions. Consequently, pattern discrimination must depend on comparisons of a vector of measurements, \mathbf{d} , not just a single value. In this section, we develop a Bayes classifier that can be applied to a pattern discrimination task.

Again, for illustrative purposes we assume that the variability is due entirely to the stimulus. Moreover, we will assume that the variability in light absorption at neighboring receptors is statistically independent. Independence means that the probability of d_i absorptions at the i^{th} receptor and d_j at the j^{th} is

$$P(d_i \& d_j | A) = P(d_i | A) P(d_j | A).$$

We can extend this principle across all of the receptors to write the probability of observing \mathbf{d} given signal A , namely

$$L_A = \prod_{i=1}^N P(d_i | A),$$

where L_A is the likelihood of observing \mathbf{d} given the stimulus A .

Finally, we must specify the spatial pattern of two stimuli. The expected number of absorptions at the i^{th} cone will depend on the spatial pattern of the stimulus. We call the mean intensity of the stimulus at location i , $\mu_{A,i}$ and $\mu_{B,i}$, respectively.

We are ready to compute the proper Bayes classifier. The general form of the decision criterion is

$$\text{If } P(\mathbf{d} | A)/P(\mathbf{d} | B) > P(B)/P(A) \text{ choose A, else B.} \quad (3.6)$$

¹In a classic paper, Hecht, Schlaer and Pirenne (1942) measured the smallest number of quanta necessary to reliably detect a signal. In the most sensitive region of the retina, under the optimized viewing conditions, they found that 100 quanta at the cornea, which corresponds to about 10 absorptions, are sufficient. The quanta are absorbed in an area covered by several hundred rods; hence, it is quite unlikely that two quanta will be absorbed by the same rod. They concluded, quite correctly, that a single photon of light can produce a measurable response in a single rod.

To find further support for their observation, Hecht et al. analyzed how the subject's responses varied across trials. They assumed that all of the observed variability was due to the stimulus, and none was due to the observer. Their account is repeated in a wonderful didactic style in Cornsweet's book. I don't think this assumption is justified; a complete treatment of the data must take into account variability intrinsic to the human observer (e.g., Nachmias and Kocher, 1970).

If the two stimuli are presented in the experiment equally often, we can re-write this equation as

$$\text{If } P(d | A) > P(d | B) \text{ choose } A, \text{ else } B. \quad (3.7)$$

Next, we can use independence to write

$$\text{If } \prod_{i=1}^N P(d_i | A) > \prod_{i=1}^N P(d_i | B) \text{ choose } A, \text{ else } B. \quad (3.8)$$

Now, we substitute the Poisson formula, take the logarithm of both sides, and regroup terms to obtain

$$\begin{aligned} \text{If } \sum_{i=1}^N d_i \ln(\mu_{A,i}/\mu_{B,i}) > \sum_{i=1}^N \mu_{A,i} - \sum_{i=1}^N \mu_{B,i} \\ \text{choose } A, \text{ else } B. \end{aligned} \quad (3.9)$$

Equation 3.9 can be interpreted as a very simple computational procedure. First, notice that the Equation contains two terms. The first term is the weighted sum of the photopigment absorptions,

$$\sum_{i=1}^N d_i w_i, \quad (3.10)$$

where $w_i = \ln(\mu_{A,i}/\mu_{B,i})$. This is a very familiar computation; it is directly analogous to the calculation implemented by a linear neuron whose receptive field sensitivity at position i is w_i (see Chapter ??).

The second term is the difference between the total number of photopigment absorptions expected from each stimulus.

$$\sum_{i=1}^N \mu_{A,i} - \sum_{i=1}^N \mu_{B,i}.$$

This term acts as a normalizing criterion to correct for the overall response to the two stimuli. The decision rule in Equation 3.9 compares a weighted sum of cone absorptions to the normalizing criterion. If the first term exceeds the second, then choose response A , else B .

We have learned that a Bayes classifier for pattern discrimination can be implemented using the results of simple linear calculations, like the calculations represented in the outputs of some peripheral neurons. In fact, making a decision like a Bayes classifier is equivalent to comparing the response of a linear neuron with a criterion value. If response of the linear neuron with receptive field defined by w_i exceeds the criterion value, then choose stimulus A , otherwise choose B .

Figure 3.2 shows an example of a Bayes classifier computed for a simple pattern discrimination task. The two spatial patterns to be discriminated are shown in panels (a) and (b). The image in (a) is the retinal image of a line segment. Figure 3.2b

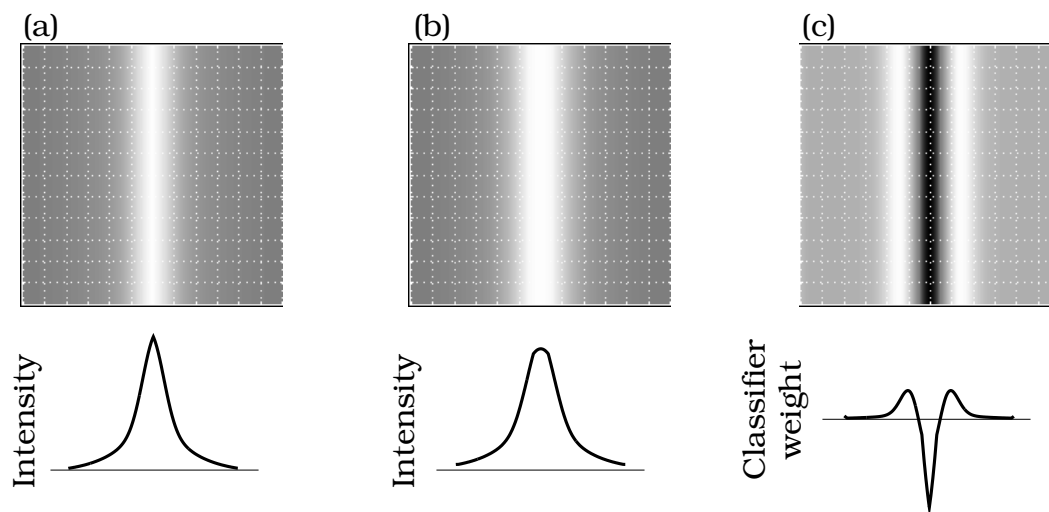


Figure 3.2: *Bayes classifier for a spatial discrimination task.* (a) The mean retinal image of a line. The grid lines are separated by 30 sec of arc, approximately the spacing of the cone inner segments in the fovea. (b) The mean retinal image of an image formed by a pair of lines separated by 30 sec of arc. The intensity of each line is one half the intensity of the line in panel (a). (c) The weights of the Bayes classifier that should be used to discriminate the stimuli in (a) and (b), given the assumptions of independent Poisson noise. (After: Geisler, 1989).

shows the retinal image of a pair of line segments, separated by 30 sec of arc. The grid marks on the image are spaced by 30 sec of arc, essentially the same spacing as the cone inner segments. The curves below the images measure the intensity of the stimuli across a horizontal section.

Figure 3.2c shows the weights of the Bayes classifier for discriminating the two images; the weights were computed using Equation 3.10. In this image the gray level represents the value of the weight that should be applied to the individual cone absorptions: a medium gray intensity corresponds to a zero weight, lighter values are positive weights, and darker values are negative weights. The spatial pattern of weights bears an obvious resemblance to the spatial patterns of receptive fields in the visual cortex.

The Bayes classifier specifies how to obtain the optimal discrimination performance when we know the signal and the noise. Apart from the optimality of the Bayes classifier itself, observers in behavioral experiments can never know the true signal and noise perfectly. The performance of the Bayes classifier, therefore, must be equal or superior to human performance. In nearly all cases where a Bayes classifier analysis has been carried out, the Bayesian classifier performance is vastly better than human observer performance. Human observers usually fail to extract a great deal of information available in the stimulus. In general, then, the Bayes classifier

does not model human performance (Banks et al., 1987).

Why, then, is the Bayes classifier analysis important? The Bayes classifier analysis defines what information is available to the observer. Defining the stimulus is an essential part of a good experimental design. The Bayes classifier defines what the observer can potentially do, and this serves as a good standard to use in evaluating performance. The Bayes classifier helps us to understand the task; only if we understand the task, can we understand the observer's performance.

The real power of this approach is that the ideal discriminator [Bayes classifier] measures all the information available to the later stages of the visual system. I would like to suggest that measuring the information available in discrimination stimuli with a model of the sort I have described here should be done as routinely as measuring the luminances of the stimuli with a photometer. In other words, we should not only use a light meter but we should also use the appropriate information meter. It is simply a matter of following the first commandment of psychophysics: "know thy stimulus." (Geisler, 1987, p. 30)

Chapter 4

Signal Estimation: A Geometric View

This book is filled with calculations of the general form

$$\mathbf{a}^T \mathbf{x} = \sum_{i=1}^{i=N} a_i x_i \quad . \quad (4.1)$$

This formula is called the *dot product* of the vectors \mathbf{a} and \mathbf{x} . It is so important that it is often given a special notation, such as $\mathbf{a} \cdot \mathbf{x}$. Every time we multiply a matrix by a vector, $\mathbf{A}\mathbf{x}$, we compute the dot product \mathbf{x} with the rows of \mathbf{A} . In this section, I want to discuss some geometric intuitions connected with the dot product operation ¹.

Although we used the dot product calculation many times, I decided not to use the dot product notation in the main portion of the book because the notation treats the two vectors symmetrically; but, in most applications the vectors \mathbf{a} and \mathbf{x} had an asymmetrical role. The vector \mathbf{x} was usually a stimulus, say the wavelength composition of a light or a one-dimensional spatial pattern, and the vector \mathbf{a} was part of the sensory apparatus, say a photopigment wavelength sensitivity or a ganglion cell receptive field. Because the physical entities we described were not symmetric, I felt that the asymmetric notation, $\mathbf{a}^T \mathbf{x}$, was more appropriate.

The scalar value of the dot product between two vectors is equal to

$$\mathbf{a} \cdot \mathbf{x} = \|\mathbf{a}\| \|\mathbf{x}\| \cos(\theta)$$

where $\|\cdot\|$ is the vector length and θ is the angle between the vectors. To simplify the discussion in the remainder of this section, we will assume that the vector \mathbf{a} has unit length.

Figure 4.1a is a geometric representation of the dot product operation. The unit vector \mathbf{a} and the signal vector \mathbf{x} are drawn as arrows extending from the origin. A

¹Physicists often refer to Equation 4.1 as the scalar product because the result is a scalar; mathematicians often refer to the dot product as an inner product and write it using angle brackets $\langle \mathbf{a}, \mathbf{x} \rangle$.

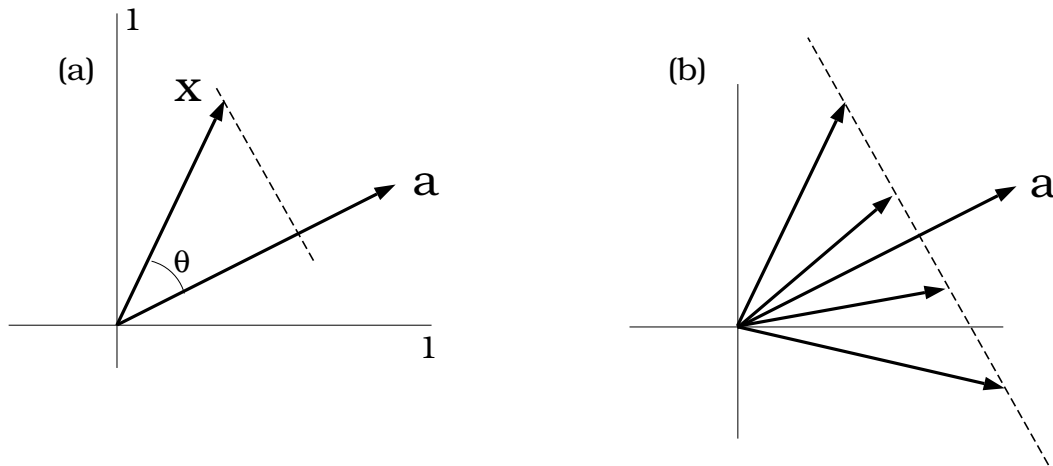


Figure 4.1: A geometric interpretation of the two-dimensional dot product. (a) A geometrical view of the dot product of two vectors is shown. The dashed line is a perpendicular from the tip of vector x to the unit-length vector a . (b) Any vector whose endpoint falls along the dashed line yields the same scalar value when we compute the vector's dot product with a .

dashed line is drawn from the tip of x (the signal) at right angles to the vector a (the sensor). The length of the vector from the origin to point of intersection between the perpendicular dashed line and a is called the *projection* of x onto a . Because we have assumed a has unit length, the length of the projection, $\|x\| \cos(\theta)$, is equal to the dot product, $a \cdot x = \|a\| \|x\| \cos(\theta)$.

By inspecting Figure 4.1b, you can see that there are many signal vectors, x , that have the same dot product with a . Specifically, all of the vectors whose endpoints fall along a line that is perpendicular to a , have the same dot product with a . Hence, any signal represented by a vector whose endpoint is on this line will cause the same response in the sensor represented by the vector a .

The geometric intuition extends smoothly to vectors with more than two dimensions. Figure 4.2a shows the dot product between a pair of three-dimensional vectors. In three-dimensions, all of the vectors whose endpoints fall on a plane perpendicular to the unit-length vector a have the same dot product with that vector (Figure 4.2b). In four or more dimensions the set of signals with a common dot product value fall on a *hyperplane*.

Figures 4.1b and 4.2b illustrate the information that is preserved and that is lost in a dot product calculation. When we measure the dot product of a pair of two-dimensional vectors, we learn that the signal must have fallen along a particular line; when we measure a three-dimensional dot-product, we learn that the signal must have fallen within a certain plane. Hence, each dot product helps to define the set of possible input signals.

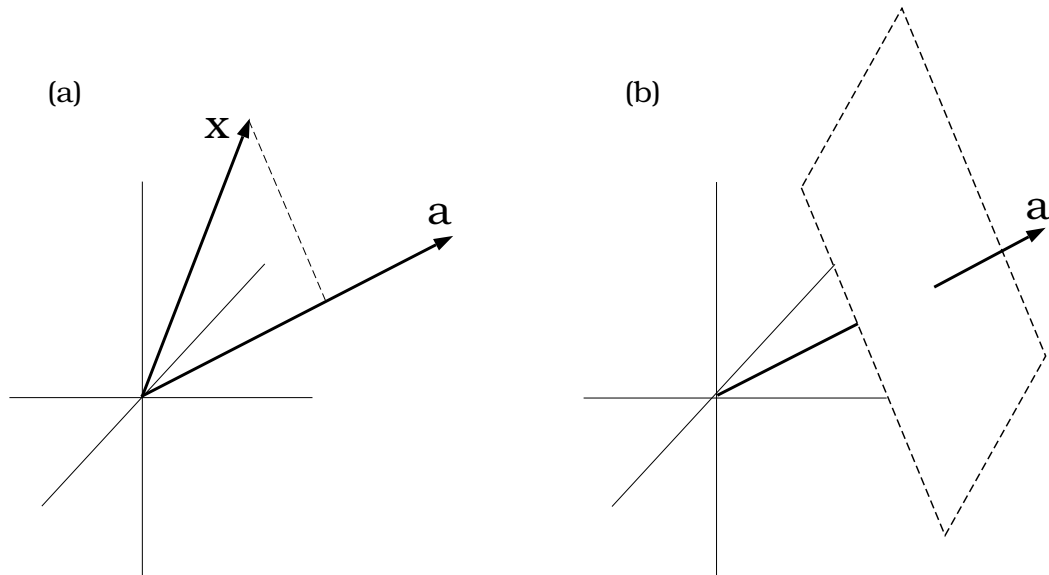


Figure 4.2: A geometric representation of the three-dimensional dot product. (a) The perpendicular from \mathbf{x} to \mathbf{a} is indicated by the dashed line. When \mathbf{a} is a unit vector, the distance from the origin to the intersection of the dashed line with \mathbf{a} is the scalar value of the dot product. (b) Any vector whose endpoint falls within the indicated plane yields the same scalar value when we compute the vector's dot with \mathbf{a} .

By combining the results from several dot products, we can estimate the input signal. We can use the geometric representation of the dot product in Figure 4.1 to develop some intuitions about how to estimate the signal from a collection of sensor responses. This is a *signal estimation* problem.

First, imagine that the response of each linear neuron is equal to the dot product of a vector describing the spatial contrast pattern of a stimulus with a vector describing the neuron's receptive field. Further, suppose that the input stimuli are drawn from a set of simple spatial patterns, namely the weighted sums of two cosinusoidal gratings,

$$w_1 \cos(2\pi f_1 x) + w_2 \cos(2\pi f_2 x)$$

We can represent each of these spatial contrast patterns using a two-dimensional vector, $\mathbf{x} = (w_1, w_2)$, whose entries are the weights of the cosinusoids. We can represent the sensitivity of each linear neuron to these spatial patterns using a two-dimensional vector, \mathbf{a}_i , whose two entries define the neuron's sensitivity to each of the cosinusoidal terms. Because the neurons are linear, we can compute the i^{th} neuron's response to any pattern in the set from the linear calculation in the dot product, namely, $\mathbf{a}_i^T \mathbf{x}$.

Figure 4.3a shows geometrically how to use two responses to identify uniquely the two-dimensional input stimulus. Suppose that the response of the neuron with receptive field \mathbf{a}_1 is r_1 . Then, we can infer that the stimulus must fall along a line

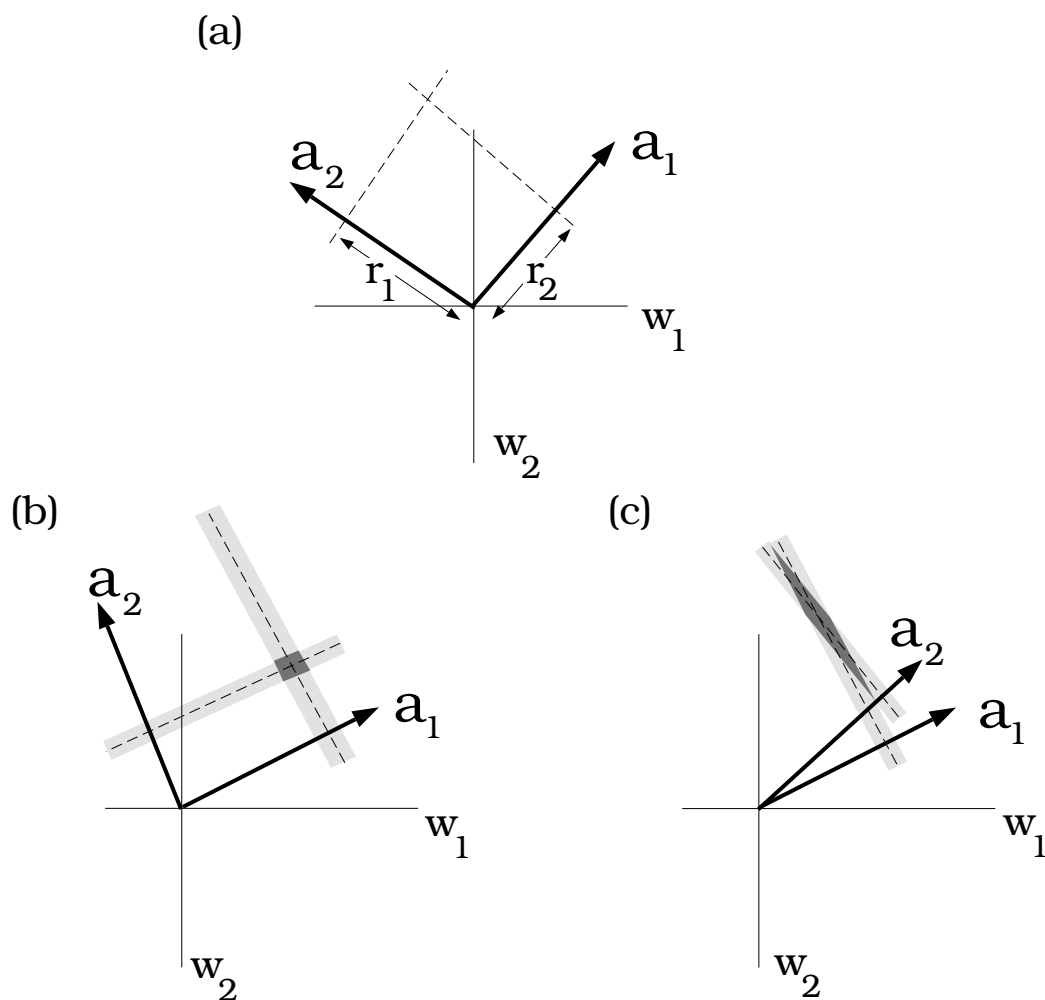


Figure 4.3: *Signal estimation from multiple linear sensors.* (a) We can infer the location of a two-dimensional signal vector from the responses of two linear sensors. (b) When the vectors representing the sensors are orthogonal, the estimation error is small. (c) When the vectors representing the sensors are nearly aligned, the estimation error tends to be quite large in the direction perpendicular to the sensor vectors.

perpendicular to the vector \mathbf{a}_1 and intersecting \mathbf{a}_1 at a distance $r_1 = \|\mathbf{x}\| \cos(\theta_1)$ from the origin². If the response to the second neuron is r_2 , we can draw a second line that describes a second set of possible stimulus locations. The true stimulus must be on both lines, so it is located at the intersection of the two dashed lines.

When the sensor responses have some added noise, which is always the case in real measurements, some sensor combinations provide more precise estimates than others. Figure 4.3b shows a pair of sensors whose vectors are nearly orthogonal to one another. Because of the sensor noise, the true identity of the signal is uncertain. The lightly shaded bands drawn around the perpendicular dashed lines show the effect of sensor noise on the estimated region from the individual measurements. The darkly shaded region is the intersection of the bands, showing the likely region containing the signal. For these orthogonal sensors, the dark band that is likely to contain the signal is fairly small.

Figure 4.3c shows the same representation but for a pair of sensors represented nearly parallel vectors. Such a pair of sensors is said to be *correlated*. When the sensors are correlated in this way, the shaded region can be quite large in the direction perpendicular to the sensor vectors. In the presence of noise, correlated sensors provide a poor estimate of the signal.

Matrix equations

Ordinarily, we use matrix multiplication to represent a collection of dot products. Many aspects of the signal estimation problem can be expressed and solved using methods developed for matrix algebra.

To see the connection between matrix multiplication and the signal estimation problem, consider the following example. Write the dot product of a two-dimensional signal and a sensor in the tableau format,

$$r_1 = (a_1 \ a_2) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

Now, suppose we have two sensors and two responses. Then, we can expand the tableau by adding more rows to represent the new measurements, as in

$$\begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad (4.2)$$

where $a_{i,\cdot}$ is a vector that represents the sensitivity of the i^{th} sensor. In the usual signal estimation problem, we know the properties of the sensors represented in the

²In general, we would place the perpendicular line at a distance of $\|\mathbf{x}\| \cos(\theta_i) / \|\mathbf{a}_i\|$, but we have assumed that \mathbf{a}_i has unit length.

rows of the matrix and we know the responses in \mathbf{r} . We wish to estimate the signal in \mathbf{x} . The pair of sensor vectors represented in the rows of the matrix, call it \mathbf{A} , are the counterpart of the vectors shown in Figure 4.3a. Our ability to estimate the signal from the responses depends on the correlation between the rows of the matrix.

The matrix tableau in Equation 4.2 shows a special case in which there are two sensors and two unknown values of the signal. In general, we might have more sensors or more unknowns in the signal. Having more sensors would force us to add more rows to the matrix; having more unknowns in the signal would force us to change the lengths of the sensor and signal vectors. Each of these changes might change the general shape of the matrix tableau. We can write the general case, without specifying the number of sensors or unknowns, using the concise representation

$$\mathbf{r} = \mathbf{A}\mathbf{x}. \quad (4.3)$$

Again, in the (linear) signal estimation problem we know the responses (\mathbf{r}) and the sensors in the rows of \mathbf{A} and we wish to estimate the signal (\mathbf{x}).

Many topics we have reviewed in the text can be framed as linear estimation problems within the matrix Equation 4.3. For example, you might imagine that the rows of the matrix are retinal ganglion receptive fields, the responses are neural firing rates, and the input is a spatial contrast pattern (Chapter ??). The brain may then need to estimate various properties of the contrast pattern from the neural firing patterns. Or, you might imagine that the rows of the matrix represent the spectral responsivities of the three cone types, the responses are the cone signals, and the signal is a spectral power distribution (Chapters ?? and ??). Or, you might imagine that the each row of the matrix represents the receptive field of a space-time oriented neuron, the output is the neuron's response, and the signal are the two velocity components of the local motion flow field (Chapter ??).

There are many matrix algebra tools that are useful for solving matrix equations like Equation 4.3. To choose the proper tool for a problem, one must take into account a variety of specific properties of the measurement conditions. For example, in some cases there are more sensors than there are unknown entries in the vector \mathbf{x} . We can represent this estimation problem using matrix tableau as

$$\begin{pmatrix} \mathbf{r} \end{pmatrix} = \begin{pmatrix} \mathbf{A} \end{pmatrix} \begin{pmatrix} \mathbf{x} \end{pmatrix}. \quad (4.4)$$

The shape of the tableau makes it plain that there are more responses in the vector \mathbf{r} than there are unknowns in the vector \mathbf{x} . This type of estimation problem is called *over-constrained*. When there is noise in the measurements, no exact solution to the over-constrained problem may exist; that is, there may be no vector \mathbf{x} such that the

equality in Equation 4.4 is perfectly satisfied. Instead, we must define an error criterion and try to find a “best” solution, that is a solution that minimizes the error criterion.

In general, the best solution will depend on the noise properties and the error criterion. In some cases, say when the noise is Gaussian and the error is the sum of the squared difference between the observed and predicted responses, there are good closed form solutions to the linear estimation problem. That is, one can find an estimate of the signal without using any search algorithms, but by direct computation. If one has other error criteria or noise, a search may be necessary.

A full discussion of the problems involved in signal estimation and matrix algebra would take us far beyond the scope of this book, but there are several excellent textbooks that explain these ideas (Strang, 1993; Lawson and Hanson, 1974, Vetterling, et al., 1992). Also, the manuals for several computer packages, such as *Matlab* and *Mathematica*, contain useful information about using these methods and further references. To see how some of these tools have been applied to vision science, you might consult some of the following references (Brainard, 1994, Heeger and Jepson, 1992; Marimont and Wandell, 1992; Nielsen and Wandell, 1988; Thomas et al., 1994; Tomasi and Kanade, 1991, Wandell, 1987)

Chapter 5

Motion Flow Field Calculation

The motion flow field describes how an image point changes position from one moment in time to the next, say, as an observer changes viewpoint. I reviewed several properties of the motion flow field in Chapter ??, and I also reviewed how to use the information in a motion flow field to estimate scene properties, including observer motion and depth maps.

In this appendix I describe how to compute a motion flow field. There are various reasons one might need to calculate a motion flow field. For example, motion flow fields are used as experimental stimuli to analyze human performance (e.g., Royden et al., 1992; Warren and Hannon, 1990). Also, algorithms designed to estimate depth maps from motion flow fields must be tested using artificial motion flow fields (e.g., Tomasi and Kanade, 1992, Heeger and Jepson, 1992).

To calculate the motion flow field, we will treat the eye as a pinhole camera and we will define the position of the points in space relative to the pinhole. Based on this framework, we will derive two formulae. First, we will derive the *perspective projection* formula. This formula describes how points in space project onto locations in the image plane of the pinhole camera. Second, we will derive how translating and rotating the pinhole camera position, i.e., the *viewpoint*, changes the point coordinates in space. Finally, we combine these two formulae to predict how changing the viewpoint changes the point locations in the image plane, thus creating the motion flow field.

Imaging Geometry and Perspective Projection

We base our calculations on ray-tracing from points in the world onto an image plane in a pinhole camera (see Chapter ??). To simplify some of the graphics calculations, it is conventional to place the pinhole at the origin of the coordinate

frame and the imaging plane in the positive quadrant of the coordinate frame. Figure 5.1a shows the geometric relationship between a point in space, the pinhole, and the image plane.

We define the pinhole to be the origin of the imaging coordinate frame. The image plane is parallel to the X-Y plane at a distance f along the Z axis. A ray from $\mathbf{p} = (p_1, p_2, p_3)$ passes towards the pinhole and intersects the image plane location (u, v, f) . Since the third coordinate, f , is the same for all points in the image plane, we can describe the image plane location using only the first two coordinates, (u, v) .

Figures 5.1bc show the geometric relationship between a point in space and its location in the image plane from two different views: looking down the Y axis and X axis, respectively. From both views, we can identify a pair of similar triangles that relate the position of the point in three space and the image point position. There are two equations that relate the point coordinates, $\mathbf{p} = (p_1, p_2, p_3)$, the distance from the pinhole to the image plane f , and the image plane coordinates (u, v) ,

$$u = (p_1/p_3)f \quad \text{and} \quad v = (p_2/p_3)f. \quad (5.1)$$

Imaging Geometry, Camera Translation and Rotation

The coordinate frame is centered at the pinhole. Hence, when the pinhole camera moves, each point in the world is assigned a new position vector. Here, we calculate the change in coordinates of the points for each motion of the pinhole optics. We will use the new coordinate to calculate the change in the point's image position, and then the motion flow field vectors.

At each moment in time we can describe the motion of the camera using two different terms. First, there is a translational component that describes the direction of the pinhole motion. Second, there is a rotational component, that describes how the camera rotates around the pinhole. We use two vectors to represent the camera's velocity. The vector $\mathbf{t} = (t_1, t_2, t_3)^T$ represents the translational velocity. Each term in this vector describes the velocity of the camera in one of the three spatial dimensions. The vector $\mathbf{r} = (r_x, r_y, r_z)^T$ represents the angular velocity. Each term in this vector describes the rotation of the camera with respect to one of the three spatial dimensions. The six values in these vectors completely describe the rigid motion of the camera.

We can compute the change in the coordinate,, $\delta\mathbf{p} = (\delta p_1, \delta p_2, \delta p_3)^T$ given the translation, rotation, and point coordinates as follows.

$$\begin{aligned} \delta p_1 &= r_z p_2 - r_y p_3 - t_1 \\ \delta p_2 &= r_x p_3 - r_z p_1 - t_2 \end{aligned}$$

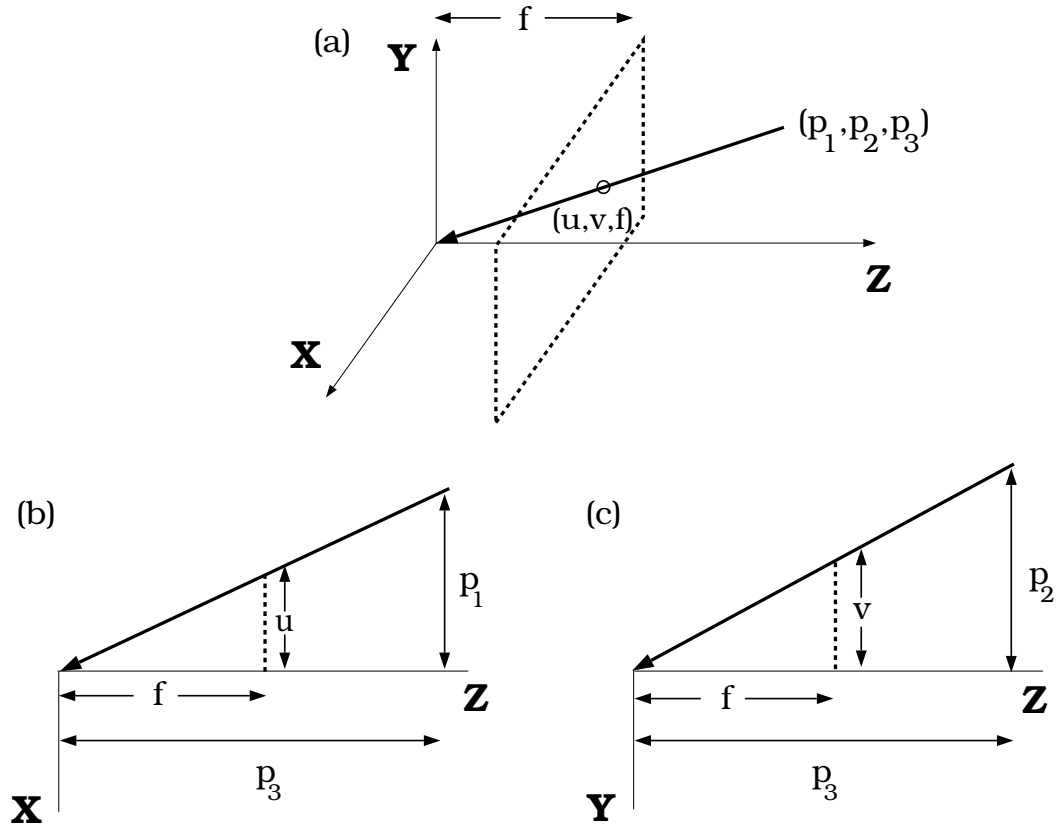


Figure 5.1: *Perspective calculations of a pinhole camera.* The coordinate frame is centered at the pinhole, and the image plane is located at a distance f in front of the pinhole parallel to the X-Y plane. (a) A ray from a point (p_1, p_2, p_3) that passes through the pinhole will intersect the image plane at location (u, v) . (b) From a view along the Y axis we find a pair of similar triangles. These triangles define an equation that relates the image plane coordinate, u , the point coordinates, and the distance from the pinhole to the image plane, f . (c) From a view along the X axis, we find an analogous pair of similar triangles and an analogous equation for v .

$$\delta p_3 = r_y p_1 - r_x p_2 - t_3$$

These formulae apply when the rotation and translation are small quantities, measuring the instantaneous change in position.

Motion Flow

Finally, we compute motion flow by using Equation 5.1 to specify how the change in position in space maps into a change in image position. The original point \mathbf{p} mapped into an image position (u, v) ; after the observer moves the new coordinate, $\mathbf{p} + \delta \mathbf{p}$ maps into a new image position, (u', v') . The resulting motion flow is the difference in the two image positions, $\mathbf{m}(u, v) = (u - u', v - v')^T$.

Equation 5.2 defines how the motion of the pinhole camera and the depth map combine to create the motion flow field (Heeger and Jepson, 1992).

$$\mathbf{m}(u, v) = \frac{1}{p_3(u, v)} \mathbf{A}(u, v) \mathbf{t} + \mathbf{B}(u, v) \mathbf{r}. \quad (5.2)$$

The term $p_3(u, v)$ is the value of p_3 for the point in three space with image position (u, v) . The entries of the 2×3 matrices $\mathbf{A}(u, v)$ and $\mathbf{B}(u, v)$ depend only on known quantities, namely the distance from the pinhole to the image plane and the image position, (u, v) .

$$\mathbf{A}(u, v) = \begin{pmatrix} -f & 0 & u \\ 0 & -f & v \end{pmatrix}$$

$$\mathbf{B}(u, v) = \begin{pmatrix} -(uv)/f & -(f + u^2/f) & v \\ (f + v^2/f) & -(uv)/f & -u \end{pmatrix}$$

Equation 5.2 expresses a relationship we have seen already: The local motion flow field is the sum of two vectors (see Figure ??). One vector describes a component of the motion flow field caused by viewpoint translation, \mathbf{t} . The second vector describes a component of the motion flow field caused by viewpoint rotation, \mathbf{r} . Only the first component, caused by translation, depends on the distance to the point, $p_3(u, v)$. The rotational component is the same for points at any distance from the viewpoint. Hence, all of the information concerning the depth map is contained in the motion flow field component that is caused by viewpoint translation.